

Learning Relations Among Movie Characters: A Social Network Perspective

Lei Ding and Alper Yilmaz

Photogrammetric Computer Vision Lab
The Ohio State University
dinglei@cse.ohio-state.edu, yilmaz.15@osu.edu

Abstract. If you have ever watched movies or television shows, you know how easy it is to tell the good characters from the bad ones. Little, however, is known “whether” or “how” computers can achieve such high-level understanding of movies. In this paper, we take the first step towards learning the relations among movie characters using visual and auditory cues. Specifically, we use support vector regression to estimate local characterization of adverseness at the scene level. Such local properties are then synthesized via statistical learning based on Gaussian processes to derive the affinity between the movie characters. Once the affinity is learned, we perform social network analysis to find communities of characters and identify the leader of each community. We experimentally demonstrate that the relations among characters can be determined with reasonable accuracy from the movie content.

1 Introduction

During recent years, researchers have devoted countless efforts on object detection and tracking to understand the scene content from motion patterns in videos [12, 7, 1, 6]. Most of these efforts, however, did not go beyond analyzing or grouping trajectories, or understanding individual actions performed by tracked objects [11, 8, 2]. The computer vision community, generally speaking, did not consider analyzing the video content from a sociological perspective, which would provide systematic understanding of the roles and social activities performed by actors based on their relations.

In sociology, the social happenings in a society are conjectured to be best represented and analyzed using a social network structure [22]. The social network structure provides a means to detect and analyze communities in the network, which is one of the most important problems studied in modern sociology. The communities are generally detected based on the connectivity between the actors in a network. In context of surveillance, a recent research reported in [23] takes advantage of social networks to find such communities. The authors use a proximity heuristic to generate a social network, which may not necessarily represent the *social structure* in the scene. The communities in the network are then detected using a common social network analysis tool referred to as the *modularity algorithm* [17]. In a similar fashion, authors of [10] generate social

relations based on the proximity and relative velocity between the actors in a scene, which are later used to detect groups of people in a crowd by means of clustering techniques.

In this paper, we attempt to construct social networks, identify communities and find the leader of each community in a video sequence from a sociological perspective using computer vision and machine learning techniques. Due to the availability of visual and auditory information, we chose to perform the proposed techniques on theatrical movies, which contain recordings of social happenings and interactions. The generality of relations among the characters in a movie introduces several challenges to analysis of the movie content: (1) it is not clear which actors act as the key characters; (2) we do not know how the low-level features relate to relations among characters; (3) no studies have been carried on how to synthesize high-level relational information from local visual or auditory cues from movies.

In order to address these challenges, our approach first aligns the movie script with the frames in the video using closed captions [5]. We should note that, the movie script is used only to segment the movie into scenes and provide a basis for generating the *scene-character relation matrix*. Alternatively, this information can be obtained using video segmentation [24] and face detection and recognition techniques [3]. A unique characteristic of our proposed framework is its applicability to an *adversarial social network*, which is a highly recognized but less researched topic in sociology [22], possibly due to the complexity of defining adversarial relations alongside friendship relations. Without loss of generality, an adversarial social network contains two disjoint rival communities $C_1 \cup C_2 = \{c_1, c_2, \dots, c_N\}$ composed of actors, where members within a community have friendly relations and across communities have adversarial relations. In our framework, we use visual and auditory information to quantify adverseness at the scene level, which serves as soft constraints among the movie characters. These soft constraints are then systematically integrated to learn inter-character affinity. The adverse communities in the resulting social network are discovered by subjecting the inter-character affinity matrix to a generalized modularity principle [4], which is shown to perform better than the original modularity [17]. Social networks commonly contain leaders who have the most important roles in their communities. The importance of an actor is quantified by computing degree, closeness, or betweenness centralities [9]. More recently, eigenvector centrality has been proposed as an alternative [19]. In this paper, due to its intrinsic relation to the proposed learning mechanism, we adopt the eigenvector centrality to find leaders in the two adverse communities. An illustration of the communities and their leaders discovered by our approach is given in Figure 1 for the movie titled *G.I. Joe: The Rise of Cobra (2009)*.

The remainder of the paper is organized as follows. We start with providing the basics of the social network framework in the next section, which is followed by a discussion on how we construct the social networks from movies in Section 3. The methodology used to analyze these social networks is described in Section



Fig. 1. Pictorial representation of communities in the movie titled *G.I. Joe: The Rise of Cobra* (2009). Our approach automatically detects the two rival communities (*G.I. Joe* and *Cobra*), and identifies their leaders (*Duke* and *McCullen*) visualized as the upscaled characters at the front of each community.

4, and is evaluated on a set of movies in Section 5. Our contributions in this paper can be summarized as follows:

- Proposal of principled methods for learning adversarial and non-adversarial relations among actors, which is new to both computer vision and sociology communities;
- Understanding these relations using a modified modularity principle for social network analysis;
- A dataset of movies, which contain scripts, closed captions and visual and auditory features, for further research in high-level video understanding.

2 Social Network Representation

Following a common practice in sociology, we define interactions between the characters in a movie using a social network structure. In this setting, the characters are treated as the vertices $V = \{v_i : v_i \text{ represents } c_i\}$ ¹ with cardinality $|V|$ and their interactions are defined as edges $E = \{(v_i, v_j) | v_i, v_j \in V\}$ between the vertices in a graph $G(V, E)$. The resulting graph G is a fully-connected weighted graph with an affinity matrix K of size $|V| \times |V|$.

In this social setting, the characters may have either adversarial or non-adversarial relations with each other. These relations can be exemplified between the characters in a war movie as non-adversarial (collaborative) within respective armies, and adversarial (competing) across the armies. Sociology and computer vision researchers often neglect adversarial relations and only analyze non-adversarial relations, such as spatial proximity relationship, friendship and kinship. The co-occurrence of both relations generates an adversarial network, which exhibits a heterogeneous social structure. Technically, adversarial or non-adversarial relation between the characters c_i and c_j can be represented by a real-valued weight in the affinity matrix $K(c_i, c_j)$, which will be decided by the proposed affinity learning method.

¹ While conventionally v is used to represent a vertex in a graph, we will also use c in this paper, and both v and c point to the same character in the movie.

A movie \mathcal{M} is composed of non-overlapping M scenes, $\mathcal{M} = s_1 \cup s_2 \cup \dots \cup s_M$, where each scene contains interactions among a set of movie characters. Appearance of a character in a scene can be encoded in a scene-character relation matrix denoted by $A = \{A_{i,j}\}$, where $A_{i,j} = 1$ if $c_j \in s_i$. It can be obtained by searching for speaker names in the script. This representation is reminiscent of the actor-event graph in social network analysis [22]. While the character relations in A can be directly used for construction of the social network, we will demonstrate later that the use of visual and auditory scene features can lead to a better social network representation by learning the inter-character affinity matrix K .

Temporal Segmentation of Movie into Scenes In order to align visual and auditory features with the movie script, we require temporal segmentation of the movie into scenes, which will provide start and stop timings for each scene. This segmentation process is guided by the accompanying movie script and closed captions. The script is usually a draft version with no time tagging and lacks professional editing, while the closed captions are composed of lines d_i , which contain timed sentences uttered by characters. The approach we use to perform this task can be considered as a variant of the alignment technique in [5] and is summarized in Figure 2:

1. Divide the script into scenes, each of which is denoted as s_i . Similarly, closed captions are divided into lines d_i .
2. Define \mathcal{C} to be a cost matrix. Compute the percentage p of the words in closed caption d_j matched with scene s_i while respecting the order of words. Set the cost as $C_{i,j} = 1 - p$.
3. Apply dynamic time warping to \mathcal{C} for estimating start t_1^i and stop times t_2^i of s_i , which respectively correspond to the smallest and largest time stamps for closed captions matched with s_i .

Due to the fact that publicly available scripts for movies are not perfectly edited, the temporal segmentation may not be precise. Regardless, our approach is robust to such inaccuracies in segment boundaries. A potential future modification of temporal segmentation can include a combination of the proposed approach with other automatic scene segmentation techniques, such as [24].

3 Learning Social Networks

Adversarial is defined “to have or involve antagonistic parties or opposing interests” between individuals or groups of people [16]. In movies or more generally in real life environments, adversarial relations between individuals are exhibited in the words they use, tones of their speech and actions they perform. Considering that a scene is the smallest segment in a movie which contains a continued event, low-level features generated from the video and audio of each scene can be used to quantify adversarial and non-adversarial contents. In the following text, we conjecture that the character members of the same community co-occur more often in non-adversarial scenes than in adversarial ones, and learn the social network formed by movie characters based on both the scene-character relations and scene contents.

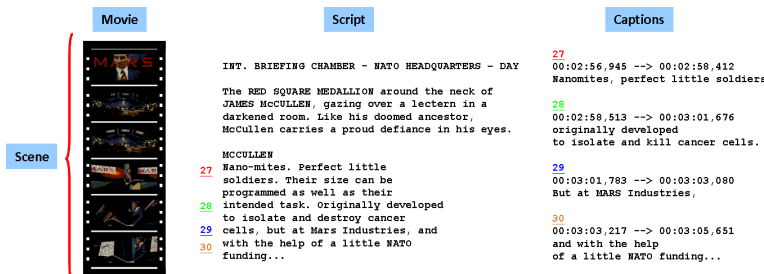


Fig. 2. Temporal segmentation of a movie into scenes. The colored numbers in the middle block indicate matched sentences in the closed captions shown on the right.

3.1 Scene Level Features and Scene Characterization

Movie directors often follow certain rules, referred to as the film grammar or cinematic principles in the film literature, to emphasize the adversarial content in scenes. Typically, adversarial scenes contain abrupt changes in visual and auditory contents, whereas these contents change gradually in non-adversarial scenes. We should, however, note that these clues can be dormant depending on the director’s style. In our framework, we handle such dormant relations by learning a robust support vector regressor from a training set.

The visual and auditory features, which quantify adversarial scene content, can be extracted by analyzing the disturbances in the video [18]. In particular for measuring visual disturbance, we follow the cinematic principles and conjecture that for an adversarial scene, the motion field is nearly evenly distributed in all directions (see Figure 3 for illustration). For generating the optical flow distributions, we use the Kanade-Lucas-Tomasi tracker [20] within the scene bounds and use good features to track. Alternatively, one can use dense flow field generated by estimating optical flow at each pixel [15]. The visual disturbance in the observed flow field can be measured by entropy of the orientation distribution as shown in Figure 4. Specifically, we apply a moving window of 10 frames with 5 frames overlapping in the video for constructing the orientation histograms of optical flows. We use histograms of optical flow vectors weighted by the magnitude of motion. The number of orientation bins is set to 10 and the number of entropy bins in the final feature vector is set to 5. As can be observed in Figure 5, flow distributions generated from adversarial scenes tend to be uniformly distributed and thus, they consistently have more high-entropy peaks compared to non-adversarial scenes. This observation serves as the basis for distinguishing the two types of scenes.

Auditory features extracted from the accompanying movie audio are used together with the visual features to improve the performance. We adopt a combination of temporal and spectral auditory features discussed in [13, 18]: energy peak ratio, energy entropy, short-time energy, spectral flux and zero crossing rate. Specifically, these features are computed for sliding audio frames that are

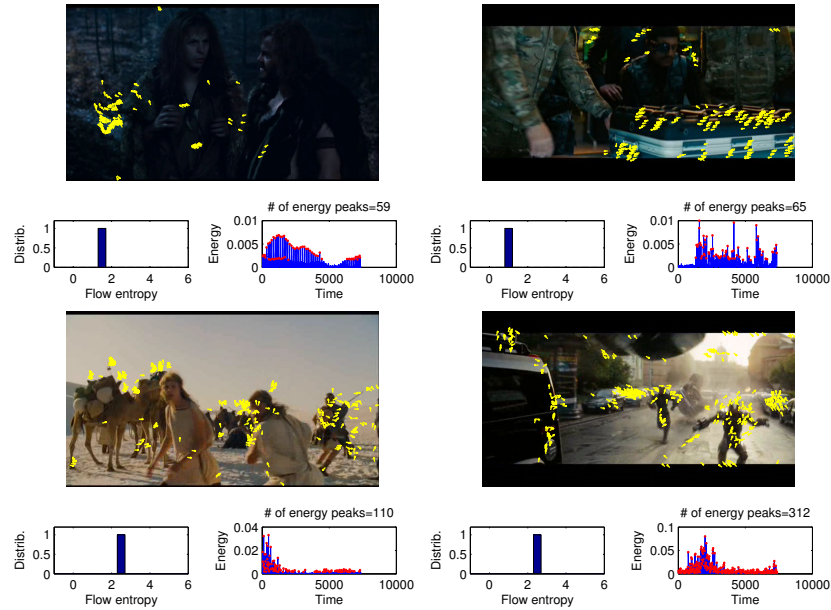


Fig. 3. Visual and auditory characteristics of adversarial scenes. Top row: non-adversarial scenes from *Year One* (2009) and *G.I. Joe: The Rise of Cobra* (2009); Bottom row: adversarial scenes from these two movies. Optical flow vectors are superimposed on the frames and computed features are shown as plots for a temporal window of 10 video frames, including entropy distribution of optical flow vectors and detected energy peaks (red dots in energy signals).

400 ms in length. The means of these features over the duration of the scene constitute a feature vector. A sample auditory feature (energy peaks) is shown in Figure 3 for both adversarial and non-adversarial scenes. It can be observed that adversarial scenes have more peaks in energy signals, which are moving averages of squared audio signals.

The visual and auditory features provide two vectors per scene (5 dimensional visual and 5 dimensional auditory), which are used to estimate a real value $\beta_i \in [-1, +1]$ for quantifying the adverseness of the scene s_i . Broadly speaking, the more negative the β_i is the more adversarial the scene is, and vice versa. In order to facilitate estimation of β_i , we use support vector regression (SVR) [21], which has been successfully used to solve various problems in recent computer vision literature. We apply a radial basis function to both the visual and auditory feature vectors, which leads to two kernel matrices \mathcal{K}_v and \mathcal{K}_a respectively. The two kernel bandwidths can be chosen by using cross-validation. The joint kernel is then computed as the multiplication kernel: $\mathcal{K} = \mathcal{K}_v \mathcal{K}_a$. Due to space limitations, we skip the details of the SVR and refer the reader to [21]. The final decision function is written as: $\beta_i = g(s_i) = \sum_{j=1}^L (\alpha_j - \alpha_j^*) \hat{\mathcal{K}}_{l,j,i} + b$, where the coefficient

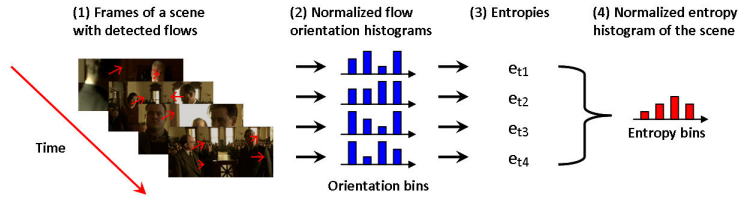


Fig. 4. Generation of the normalized entropy histogram from orientation distributions of optical flows detected from a scene.

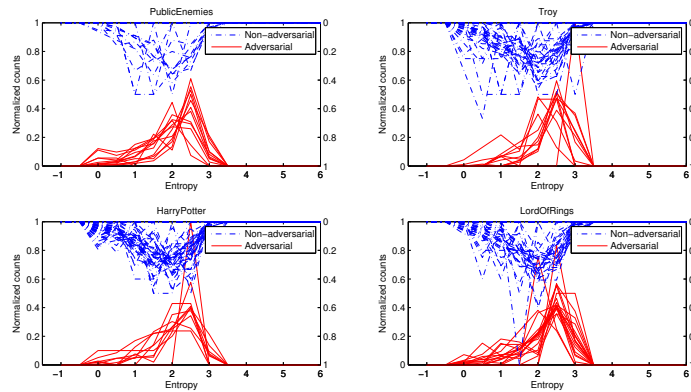


Fig. 5. Visualization of entropy histogram feature vectors extracted from four example movies. The two classes (adversarial and non-adversarial) have distinct patterns, in that adversarial scenes tend to consistently produce strong peaks in high entropies. Best viewed in color.

b is offset, α_i and α_i^* are the Lagrange multipliers for labeling constraints, L is the number of labeled examples, and l_j is the index for the j^{th} labeled example.

The training for support vector regression is achieved by using a set of scenes labeled as adversarial ($\beta_i = -1$) and non-adversarial ($\beta_i = +1$). We define a non-adversarial scene in the training and test sets as a scene which contains character members from only one group. Conversely, a scene in which the members of rival groups co-occur is labeled as adversarial. Considering that the adverseness of a scene is sometimes unclear and involves high-level semantics instead of pure observations, the stated approach avoids the subjectiveness in scene labeling. The labeling of scenes s_i in the novel movie \mathcal{M} is then achieved by estimating corresponding β_i using the regression learned from labeled scene examples from other movies in a dataset.

3.2 Learning Inter-character Affinity

Let c_i be character i , and $\mathbf{f} = (f_1, \dots, f_N)^T$ be the vector of community memberships containing ± 1 values, where f_i refers to the membership of c_i . Let \mathbf{f} distribute according to a zero-mean identity-covariance Gaussian process $P(\mathbf{f}) = (2\pi)^{-N/2} \exp^{-\frac{1}{2}\mathbf{f}^T\mathbf{f}}$. In order to model the information contained in the scene-character relation matrix A and the aforementioned adverseness of each scene β_i , we assume the following distributions: (1) if c_i and c_j occur in a non-adversarial scene k ($\beta_k \geq 0$), we assume $f_i - f_j \sim \mathcal{N}(0, \frac{1}{\beta_k^2})$; (2) if c_i and c_j occur in an adversarial scene k ($\beta_k < 0$), we assume $f_i + f_j \sim \mathcal{N}(0, \frac{1}{\beta_k^2})$.

Therefore, if $\beta_i = 0$, then the constraint imposed by a scene becomes inconsequential, which corresponds to the least confidence in the constraint. On the other hand, if $\beta_i = \pm 1$, the corresponding constraint becomes the strongest. Because of the distributions we use, none of the constraints is hard, making our method relatively flexible and insensitive to prediction errors. Applying the Bayes' rule, the posterior probability of \mathbf{f} given the constraints is defined by:

$$P(\mathbf{f}|A, \beta) \propto \exp\left(-\frac{1}{2}\mathbf{f}^T\mathbf{f} - \sum_{k:\beta_k \geq 0} \sum_{c_i, c_j \in s_k} \frac{(f_i - f_j)^2 \beta_k^2}{2} - \sum_{k:\beta_k < 0} \sum_{c_i, c_j \in s_k} \frac{(f_i + f_j)^2 \beta_k^2}{2}\right).$$

It can be verified that $P(\mathbf{f}|A, \beta) \propto \exp(-\frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f})$ is a Gaussian process with zero mean. Using $K_{i,j} = E\{f_i f_j | A, \beta\}$ as the learned affinity between c_i and c_j , it follows that $K = M^{-1}$, where

$$M_{i,j} = \begin{cases} \sum_{k:c_i, c_j \in s_k, \beta_k < 0} \beta_k^2 - \sum_{k:c_i, c_j \in s_k, \beta_k \geq 0} \beta_k^2 & \text{if } i \neq j \\ 1 + \sum_{l \neq i} \sum_{k:c_l, c_i \in s_k} \beta_k^2 & \text{if } i = j \end{cases}$$

The resulting K is symmetric and positive definite. However, unlike an affinity matrix from a Gaussian kernel, it may contain negative values. The proposed approach has two special cases:

- In the case when $\beta_i = 1$, then the aforementioned learning mechanism reduces to a co-occurrence based approach which is a traditional tool in social network analysis [17, 4]. Specifically, $M_{i,j}$, for $i \neq j$, represents the minus value of the number of scenes where c_i and c_j occur together. This reduced scheme does not utilize the video/audio feature based prediction of adverseness, and serves as a natural baseline in this paper.
- If we use fixed variance parameters in the assumed distributions instead of the learned ones, our affinity learning method reduces to the affinity propagation approach proposed in [14].

4 Social Network Analysis

In this section, we deal with grouping the movie characters into communities and finding the leader of each community. A common approach to detecting communities from a social network is to cluster vertices of the corresponding graph

using the modularity-cut [17], which has been recently used in context of surveillance [23]. For social environments, a recent study reported in [4] has shown that community detection performance of [17] can be increased by considering a generalized objective referred to as the *max-min modularity*. Their proposed algorithm, however, assumes unweighted edges and is not directly suitable for our social networks which contain weighted edges of learned strength.

In our design, we first generate a *principal affinity matrix* K' by the following rules: $K'_{i,j} = K_{i,j}$ for $K_{i,j} > 0$, and $K'_{i,j} = 0$ for other entries. We then generate a *complementary affinity matrix* K'' by the following rules: $K''_{i,j} = -K_{i,j}$ for $K_{i,j} < 0$, and $K''_{i,j} = 0$ for other entries. The matrix K'' represents the *unrelatedness* between vertices in the network in terms of community memberships. Adopting the strategy in [4] and using K' and K'' , we formulate the max-min modularity criterion as $Q_{MM} = Q_{max} - Q_{min}$ for:

$$Q_{max} = \frac{1}{2m'} \sum_{i,j} (K'_{ij} - \frac{k'_i k'_j}{2m'}) (f_i f_j + 1) \triangleq \frac{1}{2m'} \sum_{i,j} B'_{i,j} (f_i f_j + 1),$$

$$Q_{min} = \frac{1}{2m''} \sum_{i,j} (K''_{ij} - \frac{k''_i k''_j}{2m''}) (f_i f_j + 1) \triangleq \frac{1}{2m''} \sum_{i,j} B''_{i,j} (f_i f_j + 1),$$

where $m' = \frac{1}{2} \sum_{i,j} K'_{ij}$, $k'_i = \sum_j K'_{ij}$, $m'' = \frac{1}{2} \sum_{i,j} K''_{ij}$, $k''_i = \sum_j K''_{ij}$ and the term $\frac{k'_i k'_j}{2m'}$ represents the expected edge strength between the characters c_i and c_j [17]. Based on this observation, we note that $K'_{i,j} - \frac{k'_i k'_j}{2m'}$ measures how much the connection between two characters is stronger than what would be expected between them, and serves as the basis for keeping the two characters in the same community. In this formulation, the max-min modularity Q_{MM} roots from the conditions for a good network division that (1) edge strength across communities should be smaller than expected, and (2) unrelated characters within a community should be minimal. These conditions can be realized by maximizing Q_{MM} . Using standard eigen-analysis, it follows that the eigenvector \mathbf{u} of $\frac{1}{2m'} B' - \frac{1}{2m''} B''$ with the largest eigenvalue maximizes a relaxed version of Q_{MM} . The resulting eigenvector solution contains real values, and we threshold them at the 0 level to obtain the desired community memberships. That is, we let $f_i = +1$ if $u_i \geq 0$, and $f_i = -1$ if otherwise.

Once the communities in the movie are extracted, their leaders can be computed by analyzing the centrality of each character in the community. In our case, since the communities correspond to two adversarial social groups, their expected leaders relate to the *hero* or the *villain* in the movie. Let the centrality score, x_i for the i^{th} movie character be proportional to the sum of the scores of all vertices which are connected to it: $x_i = \frac{1}{\lambda} \sum_{j=1}^N K'_{i,j} x_j$, where N is the total number of characters in the movie and λ is a constant. It follows from this notation that the centralities of characters satisfy $K' \mathbf{x} = \lambda \mathbf{x}$ in the vector form. It can be shown that the eigenvector with largest eigenvalue provides the desired centrality measure [19]. Therefore, if we let the eigenvector of K' with the largest eigenvalue be \mathbf{v} , the leaders of the two communities are given by $\arg \max_{i: u_i \geq 0} v_i$ and $\arg \max_{i: u_i < 0} v_i$ respectively.

5 Experiments

For qualitative and quantitative evaluation of the proposed approach, we generate a dataset of 10 movies which contains recent and classical theatrical movies that cover a range of genres including action, adventure, fantasy and drama.² The movies in our dataset broadly contain two rival communities with a designated leader for each community. For each movie with statistics tabulated in Table 1, the dataset contains visual and auditory features, movie script, and closed caption data, all of which are temporally aligned.

Table 1. Statistics of movies in our dataset which includes the number of scenes in the movie, the number of lines in closed caption data, the total number of characters in the movie and the number of characters in one of the two communities.

Movies enumerated in footnote 2	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
# of scenes	198	151	238	226	116	51	105	297	188	199
# of captions in lines	1143	1585	1063	1155	1337	1515	1293	1262	1099	1402
# of characters in total	11	7	10	10	7	7	6	8	10	9
# of characters in community 1	6	4	6	5	3	3	4	6	7	7

In the following discussion, we analyze social networks with accompanying affinity matrices generated from

- the character co-occurrence information reflected in matrix A (co-occurrence), which is more traditional in sociology;
- in addition to co-occurrence, scene adverseness characterizations β_i which are learned from video and audio contents using the proposed approach.

In order to evaluate the contribution of these features, we provide comparisons of collective use of visual and auditory features with their individual use in extraction of communities and their leaders (details are discussed in Section 3). Due to space limitations, in Figure 6, we only show graphical representations of the social networks for ten movies learned from both visual and auditory features using the proposed approach. The color codes in the figure reflect the strength of affinity between characters. We observe that inter-community connections tend to be weaker than certain intra-community ones.

In this paper, the affinity between the characters are strongly related to the adverseness of the scenes in which they appear. This relation suggests validation of how effective the support vector regression (SVR) is for estimating the scene adverseness. In order to facilitate this, we compute the mean square error (MSE)

² The movies in our dataset are (1) *G.I. Joe: The Rise of Cobra* (2009); (2) *Harry Potter and the Half-Blood Prince* (2009); (3) *Public Enemies* (2009); (4) *Troy* (2004); (5) *Braveheart* (1995); (6) *Year One* (2009); (7) *Coraline* (2009); (8) *True Lies* (1994); (9) *The Chronicles of Narnia: The Lion, the Witch and the Wardrobe* (2005); (10) *The Lord of the Rings: The Return of the King* (2003). The dataset is available at <http://dpl.ceegs.ohio-state.edu/resources.php>.

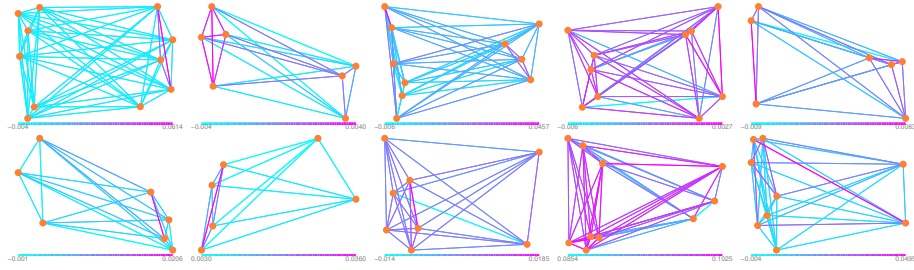


Fig. 6. Social networks generated using the proposed approach for the ten movies in our dataset. Characters (vertices) are placed on the left and right with respect to communities they belong to. The strength of affinity is indicated by pinkness of the edges: the stronger the edge is the pinker it is. Best viewed in color.

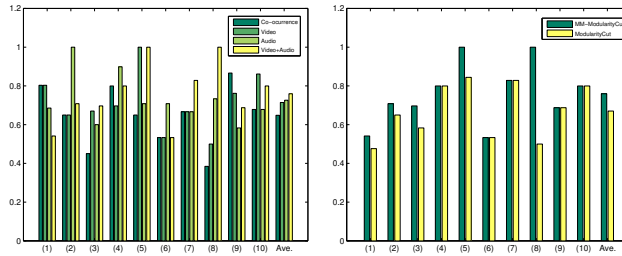


Fig. 7. Accuracy of social network analysis in $F1$ measures. Left: comparison of four approaches, where the proposed one is video+audio; Right: comparison of two modularity algorithms (max-min vs. original), with the proposed video+audio approach.

as our error measure over all the scenes in each movie, and average the resulting MSEs over all ten movies. When both the visual and auditory features are used, the MSE is estimated as 0.61. In contrast, when only one of the features is used MSE increases to 0.80 for visual only and 0.77 for auditory only. These numbers translate into accuracy rates for predicting if a scene is adversarial or non-adversarial. Respectively, the accuracy rates are computed as 81.6%, 78.2% and 78.7% for collective feature use, visual only and auditory only. These numbers reflect that the adverseness estimates of scenes can be further utilized to infer the relations among the movie characters.

The accuracy of community detection relates to how precise the assignment of the characters is into each one of the community. Considering that a community is a set of characters, the accuracy can be measured using the precision and recall values of predicted assignments given the ground truth. For each community these two values can be combined into an $F1$ measure, which is the harmonic mean of precision and recall. This measure takes into account the possible imbalance in the size of communities and has been widely adopted. Considering

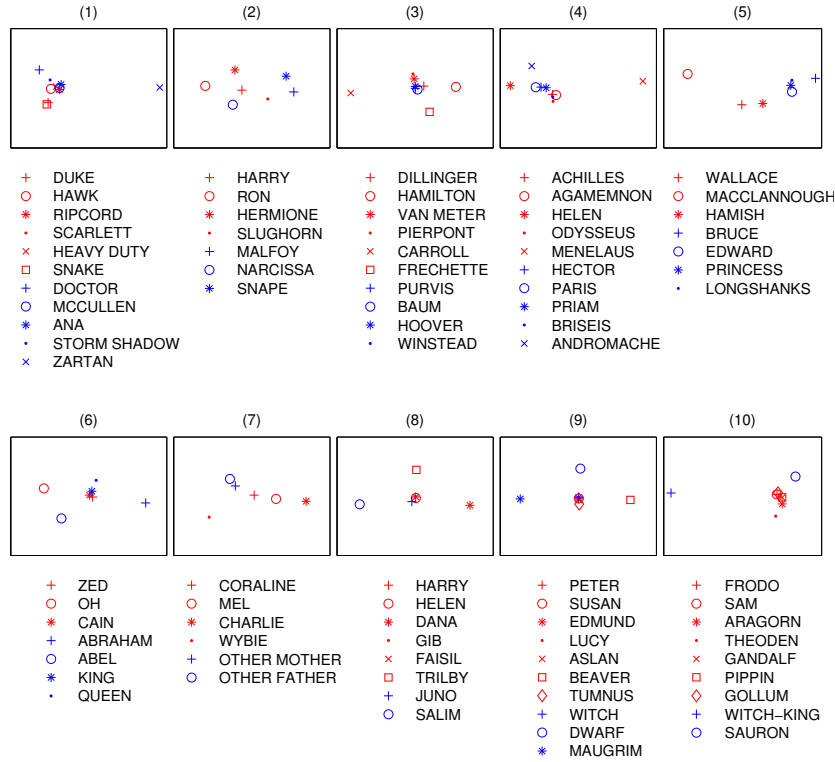






















Fig. 8. 2D visual maps of character relations. Red and blue stand for the two communities respectively according to our ground truth labeling. Best viewed in color.

that the movies in our dataset contains more than one community, we report the average $F1$ measure over detected communities as the final detection accuracy for each movie. From the quantitative evaluations shown in Figure 7, for four movies visual features help enhance performance appreciably. Overall, auditory features improve the performance slightly more than the visual features when they are used independently. Their combination, however, provides the best performance, which on the average leads to an $F1$ measure of 76.0%. This score, when compared to using only the character co-occurrence to generate the social network, improves the grouping performance by 11.1%. In the same figure, we also show that the modified max-min modularity, when compared to the traditional modularity computed from K , improves the $F1$ measure by 8.9%.

As discussed in Section 4, the community assignment of characters is realized by analyzing the eigenspace of $\frac{1}{2m'}B' - \frac{1}{2m''}B''$. In order to visualize this assignment process, we map the characters in the movie into coordinates defined by the two eigenvectors with highest eigenvalues. This mapping provides an optimal way to visualize the inter-character relations in two dimensions. In the

Table 2. Community leaders discovered using the proposed framework. The names in bold face refer to correct ones, whereas those in italics are not.

Movies	(1)	(2)	(3)	(4)	(5)
Community 1	 <i>Hawk</i>	 Harry	 Dillinger	 <i>Achilles</i>	 <i>MacClan.</i>
Community 2	 McCullen	 Snape	 Purvis	 <i>Androm.</i>	 Longsha.
Movies	(6)	(7)	(8)	(9)	(10)
Community 1	 Zed	 Coraline	 <i>Trilby</i>	 <i>Susan</i>	 Frodo
Community 2	 <i>Abraham</i>	 OtherMo.	 Salim	 Witch	 WitchKing

figure, we illustrate the ground truth in red and blue colors respectively for the two communities³. As can be observed, the characters who belong to separate communities tend to lie apart.

As discussed in Section 4, the eigenvector of K' with the highest eigenvalue provides the leaders of communities. In Table 2, we tabulate these leaders with their pictures for the two rival communities for each movie. The predicted leaders who correspond to the true leaders in the movie are shown in bold face, while incorrect leaders are shown in italics. Overall, it can be observed that many of the leaders are successfully discovered by our framework.

6 Conclusions and Future Work

In this paper, we have presented the first work on learning the relations among characters from movies using a social network approach. We have used visual and auditory features to characterize the adverseness of each scene in a movie. By using an affinity learning procedure, we incorporate the scene adverseness, and make informed decisions in constructing and analyzing the corresponding social network. Extensive analysis on a set of 10 movies has validated the effectiveness of our framework in high level understanding of social interactions. The proposed framework also contributes to sociology by leveraging computer vision and machine learning techniques. Although we present our framework on analysis of movies, it is possible to apply it to other problem domains, such as video surveillance, in which suspicious behaviors can be related to the interactions between objects in a scene.

³ In movie (10), *Gollum* has a good personality except for when he is close to *the ring*. The ring changes the good behavior of the characters to bad except for *Frodo*.

References

1. Ali, S., Basharat, A., Shah, M.: Chaotic invariants for human action recognition. In: ICCV (2007)
2. Alon, J., Athitsos, V., Yuan, Q., Sclaroff, S.: A unified framework for gesture recognition and spatiotemporal gesture segmentation. *IEEE Trans. on PAMI* 31(9), 1685–1699 (2009)
3. Arandjelović, O., Zisserman, A.: Automatic face recognition for film character retrieval in feature-length films. In: CIVR (2005)
4. Chen, J., Zaiane, O., Goebel, R.: Detecting communities in social networks using max-min modularity. In: SDM (2009)
5. Cour, T., Jordan, C., Mitsakaki, E., Taskar, B.: Movie/script: Alignment and parsing of video and text transcription. In: ECCV (2008)
6. Ding, L., Fan, Q., Hsiao, J., Pankanti, S.: Graph based event detection from realistic videos using weak feature correspondence. In: ICASSP (2010)
7. Efros, A.A., Berg, A.C., Mori, G., Malik, J.: Recognizing action at a distance. In: ICCV (2003)
8. Fathi, A., Mori, G.: Action recognition by learning mid-level motion features. In: CVPR (2008)
9. Freeman, L.: Centrality in social networks: Conceptual clarification. *Social Networks* 1(3), 215–239 (1979)
10. Ge, W., Collins, R., Ruback, B.: Automatically detecting the small group structure of a crowd. In: WACV (2009)
11. Jiang, H., Fels, S., Little, J.: A linear programming approach for multiple object tracking. In: CVPR (2007)
12. Laptev, I., Lindeberg, T.: Space-time interest points. In: ICCV (2003)
13. Lin, J., Wang, W.: Weakly-supervised violence detection in movies with audio and video based co-training. In: PCM (2009)
14. Lu, Z., Carreira-Perpinan, M.A.: Constrained spectral clustering through affinity propagation. In: CVPR (2008)
15. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: IJCAI (1981)
16. Merriam-Webster: Merriam-webster online dictionary. <http://www.merriam-webster.com/dictionary> (2010)
17. Newman, M.E.J.: Modularity and community structure in networks. *PNAS* 103(23), 8577–8582 (2006)
18. Rasheed, Z., Shah, M.: Movie genre classification by exploiting audio-visual features of previews. In: ICPR (2002)
19. Ruhnau, B.: Eigenvector-centrality? a node-centrality. *Social Networks* 22(4), 357–65 (2000)
20. Shi, J., Tomasi, C.: Good features to track. In: CVPR (1994)
21. Smola, A.J., Schölkopf, B.: A tutorial on support vector regression. *Statistics and Computing* 14(3), 199–222 (2004)
22. Wasserman, S., Faust, K., Iacobucci, D.: *Social Network Analysis: Methods and Applications*. Cambridge University Press (1994)
23. Yu, T., Lim, S.N., Patwardhan, K., Krahnstoeber, N.: Monitoring, recognizing and discovering social networks. In: CVPR (2009)
24. Zhai, Y., Shah, M.: Video scene segmentation using markov chain monte carlo. *IEEE Trans. on Multimedia* 8(4), 686–697 (2006)